We claim:

1.   A method for processing OpenGL display list commands issued by a client, comprising:

providing a daemon and an OpenGL stack;

intercepting display list commands issued by the client to the stack;

responsive to an intercepted newList [list commands] endList command sequence:

    storing, in the daemon, a representation of a first subset of the list commands; and

    creating, in the stack, a display list corresponding to a second subset of the list commands; and

responsive to an intercepted callList command, causing the daemon to issue to the stack:

    at least one immediate mode command corresponding to the first subset of the list commands; and

    at least one callList command invoking the display list corresponding to the second subset of the list commands.

2.   The method of claim 1, further comprising:

responsive to the intercepted newList [list commands] endList command sequence:

    storing, in the daemon, a representation of an expression or function; and

responsive to the intercepted callList command, causing the daemon to:

    evaluate the expression or function; and

    depending on the value of the expression or function, conditionally change the step of invoking the display list.

3.   The method of claim 2, wherein:

creating the display list comprises creating plural display lists corresponding to different levels of detail;

evaluating the expression or function yields a level of detail indicator; and

conditionally changing the step of invoking the display list comprises invoking, from among the plural display lists, the one display list whose level of detail corresponds to the level of detail indicator.

4.   The method of claim 2, wherein:

evaluating the expression or function yields a visibility indicator; and

conditionally changing the step of invoking the display list comprises invoking or not invoking the display list depending on the value of the visibility indicator.

5.   The method of claim 4, wherein:

the visibility indicator represents whether a pre-defined bounding volume intersects a viewing volume.

6.   Apparatus for processing OpenGL display list commands issued by a client, comprising:

daemon logic and an OpenGL stack;

means for intercepting display list commands issued by the client to the stack;

means for responding to an intercepted newList [list commands] endList command sequence, comprising:

> means for storing, in the daemon, a representation of a first subset of the list commands; and

> means for creating, in the stack, a display list corresponding to a second subset of the list commands; and

means for responding to an intercepted callList command by causing the daemon to issue to the stack:

> at least one immediate mode command corresponding to the first subset of the list commands; and

> at least one callList command invoking the display list corresponding to the second subset of the list commands.

7.   The apparatus of claim 6, further comprising:

means for responding to the intercepted newList [list commands] endList command sequence by storing, in the daemon, a representation of an expression or function; and

means for responding to the intercepted callList command comprising:

> means for evaluating the expression or function; and

> means for, depending on the value of the expression or function, conditionally changing the step of invoking the display list.

8.   The apparatus of claim 7, wherein:

the means for creating the display list comprises means for creating plural display

lists corresponding to different levels of detail;

the means for evaluating the expression or function yields a level of detail indicator;

and wherein

conditionally changing the step of invoking the display list comprises invoking, from

among the plural display lists, the one display list whose level of detail

corresponds to the level of detail indicator.

9.   The apparatus of claim 7, wherein:

the means for evaluating the expression or function yields a visibility indicator; and

wherein

conditionally changing the step of invoking the display list comprises invoking or not

invoking the display list depending on the value of the visibility indicator.

10.   The apparatus of claim 9, wherein:

the visibility indicator represents whether a pre-defined bounding volume intersects

a viewing volume.

11.   Program code embodied in a storage or transmission medium which, when executed by one or more processing devices, implements a method for processing OpenGL display list commands issued by a client, the method comprising:

providing a daemon and an OpenGL stack;

intercepting display list commands issued by the client to the stack;

responsive to an intercepted newList [list commands] endList command sequence:

      storing, in the daemon, a representation of a first subset of the list commands;
         and

      creating, in the stack, a display list corresponding to a second subset of the list commands; and

responsive to an intercepted callList command, causing the daemon to issue to the stack:

      at least one immediate mode command corresponding to the first subset of the list commands; and

      at least one callList command invoking the display list corresponding to the second subset of the list commands.

12.   The program code of claim 10, wherein the method further comprises:

responsive to the intercepted newList [list commands] endList command sequence:

      storing, in the daemon, a representation of an expression or function; and

responsive to the intercepted callList command, causing the daemon to:

      evaluate the expression or function; and

      depending on the value of the expression or function, conditionally change the step of invoking the display list.

13.   The program code of claim 12, wherein:

creating the display list comprises creating plural display lists corresponding to
        different levels of detail;

evaluating the expression or function yields a level of detail indicator; and

conditionally changing the step of invoking the display list comprises invoking, from
        among the plural display lists, the one display list whose level of detail
        corresponds to the level of detail indicator.

14.   The program code of claim 12, wherein:

evaluating the expression or function yields a visibility indicator; and

conditionally changing the step of invoking the display list comprises invoking or not
        invoking the display list depending on the value of the visibility indicator.

15.   The program code of claim 14, wherein:

the visibility indicator represents whether a pre-defined bounding volume intersects
        a viewing volume.